team FLEX!

. . . dream . . . design . . . deliver . . .

Arduino Uno with Flex

Team:

Stacy Bridges, Timothy Case, and Kelly Andreola

Table of Contents

- Introduction 3
- The Microcontroller Platform 3
- The Test Device 4
- Development Tools 4
- Your Experiment 5
- Conclusions 6
- Contributions 6
- Project Code 7
- References 8

Introduction

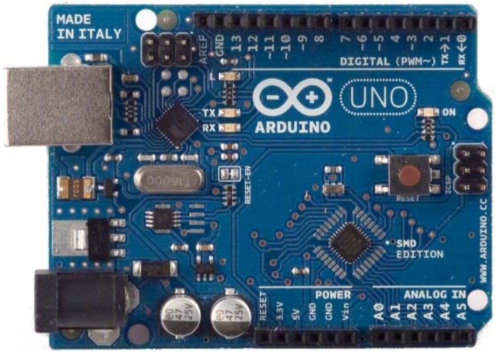
Team Flex! is made up of the Arduino and flex sensor. Our team decided to make a theremin-inspired device using those pieces. In order to do this, we used many things found in a basic Arduino kit (e.g., LEDs, jumpers, photoresistors, speakers, a breadboard, a buzzer, resistors, etc.).

The Microcontroller Platform

The Arduino is based on the ATmega328. The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

The microcontroller board has over 10 digital pins, 6 analog pins, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. The Arduino Uno, specifically, uses the Atmega16U2 (Atmega8U2 up to version R2) as a USB-to-serial driver chip instead of the FTDI USB-to-serial driver chip. It can be found online, or in stores like RadioShack and Fry’s.

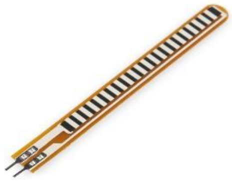
If any specific part happened to be more useful for our chosen project, then the ability to utilize analog input was essential.



The Test Device

Our project used the flex sensor to control the pitch of our output via speakers. As the sensor is flexed, the resistance across the sensor increases. Connector is 0.1" spaced and breadboard friendly.

The flex sensor can be found online (at places like SparkFun and Adafruit).



Development Tools

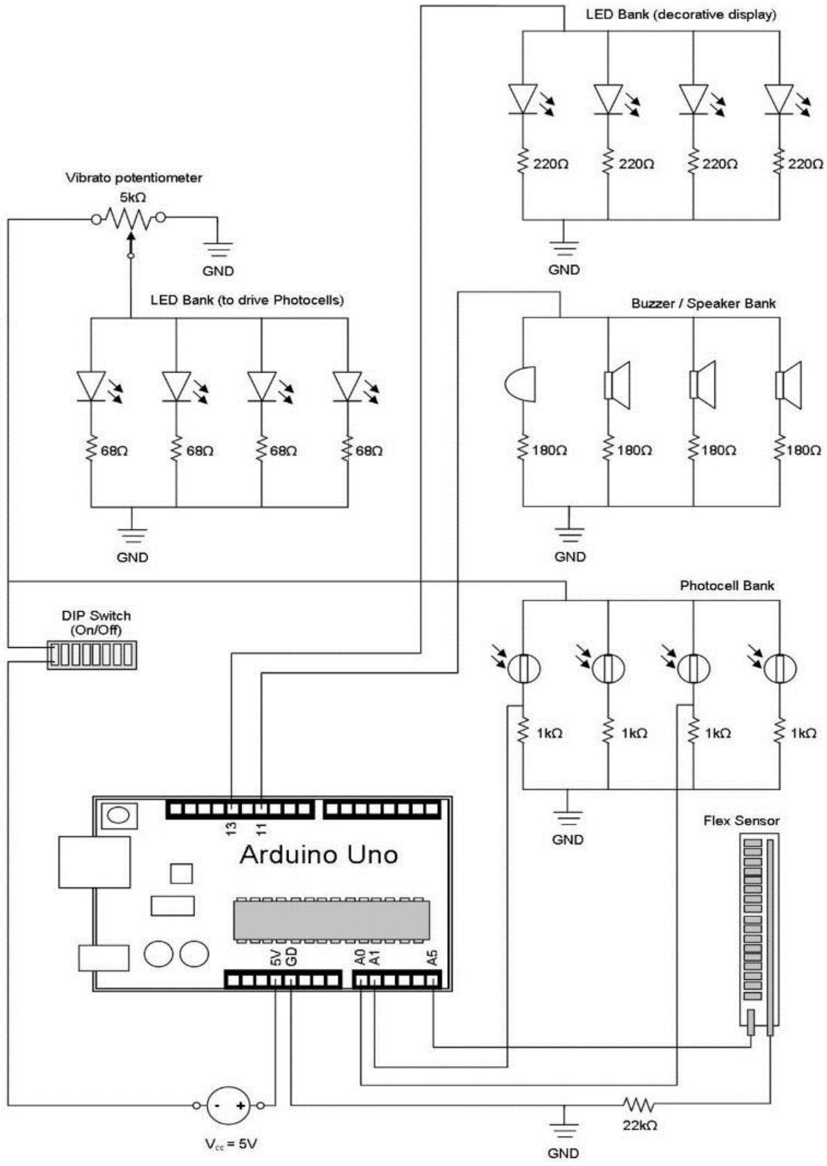
Below is a detailed list of the tools we obtained to work with the microcontroller and the device. We also used the Arduino software for our project.

Part Description	Qty
Bright White LEDs	5
Bright Red LED	1
Bright Green LED	1
Bright Yellow LED	1
Photo cells, Cadmium Sulphide (various resistance)	4
Mini Speakers (8 Ohm, 0.1W)	3
Piezo Buzzer	1
Resistor, Carbon Film (0.5W, 68Ω)	4
Resistor, Carbon Film (0.5W, 181Ω)	4
Resistor, Carbon Film (0.5W, 220Ω)	4
Resistor, Carbon Film (0.5W, 1kΩ)	4
Resistor, Carbon Film (0.5W, 22kΩ)	1
Dual-Inline Package (DIP) Switch	1
Flex Sensor (4.4", 0.5W, 10kΩ-110kΩ)	1
Potentiometer (Linear, 5kΩ)	1

Your Experiment

As a team, we were drawn toward a project involving sound, robotics, and/or a light show. We eventually discussed the theremin concept in one of our brainstorming sessions, which ended up being the direction we chose for our project.

Our team used a switch as a power on/off device, and while that switch was on, there was a dimmer switch that powered the LEDs that, in turn, powered the photoresistor cells. Then the output from the photoresistors was used in order to set thresholds for the display LEDs (i.e., the blinking) and the vibrato function of the speakers. The flex sensor was utilized to affect the pitch of the speakers uniquely from the vibrato/blinking function.
Schematic:



Conclusions

Kelly – Since I had never touched an Arduino or flex sensor, or even a breadboard, I couldn't list all of the things I've learned. Probably the most important thing I learned was about electricity flow. Learning how to set up breadboards was very interesting (and challenging) for me. I'd really like to extend what our project did, and make gloves that have multiple flex sensors (which has already been done), but have each sensor connected to a different instrument sound. I'd like to go in a different direction, but Imogen Heap is known for the music gloves they've created (http://www.imogenheap.co.uk/thegloves/).

Stacy - I learned that the Arduino has a limited amount of power that it can output to the project components. It can provide approximately 1/2 amp to all of the outputs combined if using USB power from the PC, or it can provide up to 1 amp to all outputs combined if using a DC power source. Within these power constraints, the Arduino is further constrained by a recommended maximum of 40 milliamps per pin, except on the voltage pin itself, which can provide up to 200 milliamps. As a result of these limitations, each Arduino project that will utilize more than just a few components should be designed with power constraints in mind. To this end, the designer should consider which pins will be used, how the power will be sourced, and whether additional hardware may be required, such as additional boards, power supplies, signal amplifiers, and so forth.

In regard to future projects, I thought it would be nice to revisit this project on a larger scale, perhaps by adding some additional tone-shaping features, some home-made speakers, a permanent project box, and a formal instruction set. The instruction set could be posted with a demo to www.instructables.com or perhaps to a professional homepage as a means of proving proficiency in the area of micro-controllers.

Timothy - I learned tons of stuff about breadboards and Arduinos. For a future project, I wouldn't change anything. Just add more features.

Contributions

Stacy – Workshop over Arduino and breadboard.
Kelly – Putting together Project Report.
Tim – Hardware setup and storage.
All – Design, testing and software writing.

Project Code

```
/***** teamFLEX! Flex Sensor Theremin *****/

// global variables

     int toSpkrs = 11; // ~11, OUTPUT
     int toLEDs = 13; // 13, OUTPUT
     int LP_0 = 0; // A0, INPUT
     int LP_1 = 1; // A1, INPUT
     int flexPot = 5; // A5, INPUT

void setup()
{
     pinMode( 11, OUTPUT );
     pinMode( 13, OUTPUT );
     Serial.begin( 9600 );
}

void loop()
{
// get reading
     int readFlex = analogRead( flexPot )/4;
     int readLP_0 = analogRead( LP_0 )/2;
     int readLP_1 = analogRead( LP_1 )/2;

// vibrato
     int vibratoUp = readFlex + 40;
     int vibratoDn = readFlex - 40;

// manipulate spkrs
     if( readFlex < 4)
     {
          noTone( toSpkrs );
          digitalWrite( toLEDs, LOW );
     }
     else if( readLP_0 > 75 && readLP_0 < 150 )
     {
          tone( toSpkrs, 200 + vibratoUp );
          digitalWrite( toLEDs, HIGH );
          delay( 80 );
```

```
        tone( toSpkrs, 200 + vibratoDn );
        digitalWrite( toLEDs, LOW );
        delay( 80 );
    }
    else if( readLP_0 > 150 )
    {
        tone( toSpkrs, 200 + vibratoUp );
        digitalWrite( toLEDs, HIGH );
        delay( 40 );
        tone(toSpkrs, 200 + vibratoDn );
        digitalWrite( toLEDs, LOW );
        delay( 40 );
    }
    else
    {
        tone( toSpkrs, 200 + readFlex );
        digitalWrite( toLEDs, HIGH );
    }

    Serial.println( readLP_0 );
}
```

References

http://arduino.cc/en/Main/arduinoBoardUno

https://www.sparkfun.com/products/10264